Theoretical Computer Science 39 (1985) 27-45
North-Holland

A PROOF-THEORETIC CHARACTERIZATION OF OBSERVATIONAL EQUIVALENCE*

Colin STIRLING

Department of Computer Science, University of Edinburgh, Edinburgh EH8 9YL, Scotland (United Kingdom)

Abstract. Hennessy and Milner have shown that observational equivalence can be characterized by a modal language. We show here that a subset of this language, without an explicit negation operator, also characterizes it. Using this subset we offer sound and complete modal proof systems for simple nondeterministic languages of processes. In the case where observational equivalence is not a congruence we show that the observational congruence can also be characterized modally. Using this new language we again offer a sound and complete proof system. We briefly comment upon proof systems for extensions of the nondeterministic process languages.

Introduction

The meaning of a nondeterministic and concurrent program cannot be given simply as a function from input to output because the program may interact or communicate repeatedly with its environment. In [8], an alternative is proposed based on the idea of an observer attempting to communicate or experiment with a program. The authors define observational equivalence between two programs in terms of indistinguishability by observers. They then offer a family of simple languages of processes expressing finite behaviours. For some of these languages observational equivalence is not a congruence: program contexts do not preserve the equivalence. So the authors define observational congruence as the largest congruence contained in the observational equivalence and show that it can be axiomatized algebraically.

Hennessy and Milner show that observational equivalence can be characterized by a modal language. (More general modal characterization results are contained in [2, 10, 12].) Here we show that a subset of this language, without an explicit negation operator, also characterizes it. Using this language we offer sound and complete modal proof systems for simple nondeterministic languages of processes. In the case where observational equivalence is not a congruence we show that the observational congruence can also be characterized modally. Using this new language we again offer a complete and sound proof system. We briefly comment upon proof systems for extensions of the nondeterministic languages. This, we believe, is a tentative first step towards a modal proof theory for Milner's Calculus of Communicating Systems [11, 13].

* This work was supported by the Science and Engineering Research Council of the U.K.

The paper consists of six sections. Proofs of the lemmas and theorems are contained in Appendix A. Section 1 recalls the definition of observational equivalence and Section 2 its modal characterization. In Section 3 we show that a sublanguage of this modal language also characterizes the equivalence. Using this language which we call L we offer in Section 4 a sound and complete proof system NL for a simple nondeterministic language of processes. For this process language observational equivalence is a congruence. The addition of unobservable actions to it changes this. So in Section 5 we show that an extension of L, L', characterizes the congruence. We also offer two sound and complete modal proof systems: UNL on L and UNL' on L'. Finally, in Section 6 we remark upon proof systems for extensions of the nondeterministic process languages.

1. Observational equivalence

A nondeterministic and concurrent program may communicate repeatedly with its environment. Therefore, the meaning of such a program cannot be given simply as an input/output function. In [8] the authors propose an alternative based upon an idea of observational indistinguishability. Let P be a set of processes and E a set of atomic experiments. An atomic experiment on $p \in P$ is to be understood as an attempt to communicate with p. Communication may change a process depending upon its internal structure. Hennessy and Milner capture the effect of an experiment using a binary relation over P: for each $e \in E$, let $R_e \subseteq P \times P$. Using these atomic experiments a sequence of equivalence relations S_n over P is defined:

$$p\, S_0\, q \quad \text{if} \quad p, q \in P,$$

$$p\, S_{n+1}\, q \quad \text{if} \quad \text{(i)} \quad \forall e \in E\ \forall p' : \langle p, p' \rangle \in R_e \text{ implies } \exists q' : \langle q, q' \rangle \in R_e \text{ and } p'\, S_n\, q',$$

$$\text{(ii)} \quad \forall e \in E\ \forall q' : \langle q, q' \rangle \in R_e \text{ implies } \exists p' : \langle p, p' \rangle \in R_e \text{ and } p'\, S_n\, q'.$$

Then p is said to be *observationally equivalent* to q, written as $p\, S\, q$, whenever $p\, S_n\, q$ for every n.

A computation can be considered as a sequence of experiments. Clearly, for two processes to be observationally equivalent sameness of computations is not enough. Observational equivalence imposes strong connections between their respective intermediate states. Such connections are, in general, needed for comparing the behaviours of concurrent programs.

2. A modal characterization of observational equivalence

Hennessy and Milner define a modal language which characterizes observational equivalence assuming that each R_e is image finite: the set $\{q : \langle p, q \rangle \in R_e\}$ is finite for

each $p \in P$ and $e \in E$ [8]. Let *the language* J be the least set such that:

$$\mathbf{true} \in J,$$

$$\neg A, \langle e \rangle A \in J \quad \text{whenever } A \in J \text{ and } e \in E,$$

$$A \wedge B \in J \quad \text{whenever } A, B \in J.$$

The authors define a *satisfaction relation* $\models_J \subseteq P \times J$ as the least relation such that:

$$p \models_J \mathbf{true} \quad \forall p \in P,$$

$$p \models_J A \wedge B \text{ iff } p \models_J A \text{ and } p \models_J B,$$

$$p \models_J \neg A \text{ iff } p \not\models_J A,$$

$$p \models_J \langle e \rangle A \text{ iff } \exists p' : \langle p, p' \rangle \in R_e \text{ and } p' \models_J A.$$

true stands for true which every process satisfies. $p \models_J \langle e \rangle$ **true** means that an e-experiment can be carried out successfully on p. More generally, $p \models_J \langle e \rangle A$ means that p can evolve under some e-experiment to a process satisfying A. Let $[e]$ be the dual of $\langle e \rangle$, $[e] = \neg\langle e \rangle\neg$, then $p \models_J [e]A$ means that every process which is the result of a successful e-experiment on p satisfies A. In particular, because no process can satisfy $\neg$**true**, $p \models_J [e]\neg$**true** means that p is e-deadlocked: no e-experiment on p can be successful.

Let $J(p) = \{A : p \models_J A\}$. Hennessy and Milner prove the following theorem.

Theorem 2.1. *If each* R_e *is image finite, then* $J(p) = J(q)$ *iff* $p \, S \, q$.

The proof of this theorem uses modal degree: the *modal degree* of a formula is the maximum depth of modal operators occurring in it. Let J_n be the sublanguage of J containing formulas whose modal degree is at most n. Then the authors show that $J_n(p) = J_n(q)$ iff $p \, S_n \, q$. The assumption that each R_e be image finite can be discarded if infinite conjunctions are allowed [10, 12].

3. A further modal characterization of observational equivalence

In this section we offer a subset of J which also characterizes observational equivalence. The subset dispenses with an explicit negation operator: this considerably aids the development of the modal proof theories provided in the sequel.

Let L be the language which is the least set such that:

$$\mathbf{true}, \mathbf{false} \in L,$$

$$A \wedge B, \; A \vee B \in L \quad \text{whenever } A, B \in L,$$

$$\langle e \rangle A, \; [e]A \in L \quad \text{whenever } A \in L \text{ and } e \in E.$$

The satisfaction relation $\models_L$ is similar to $\models_J$:

$p \models_L \mathbf{true}$ $\forall p \in P$,

$p \not\models_L \mathbf{false}$ $\forall p \in P$,

$p \models_L A \wedge B$ iff $p \models_L A$ and $p \models_L B$,

$p \models_L A \vee B$ iff $p \models_L A$ or $p \models_L B$,

$p \models_L \langle e \rangle A$ iff $\exists p' : \langle p, p' \rangle \in R_e$ and $p' \models_L A$,

$p \models_L [e]A$ iff $\forall p' : \langle p, p' \rangle \in R_e$ implies $p' \models_L A$.

L is a sublanguage of J where **false** is defined as $\neg$**true**, $A \vee B$ as $\neg(\neg A \wedge \neg B)$ and $[e]A$ as before. L characterizes observational equivalence because $L \subseteq J$ and every formula of J is semantically equivalent to some formula of L: where A, B are formulas then A is semantically equivalent to B, which we write as $A \equiv B$, just in case $\forall p \in P$: $p \models A$ iff $p \models B$; the satisfaction relations in this definition will be relativized to whatever languages A and B belong to. This semantic relationship holds between J and L because negations can be 'moved inwards' in any formula of J using the equivalences $\neg$**true** $\equiv$ **false**, $\neg\neg A \equiv A$, $\neg(A \wedge B) \equiv \neg A \vee \neg B$, and $\neg\langle e \rangle A \equiv [e]\neg A$. Consequently, the following theorem holds.

Theorem 3.1. $\forall A \in J, \exists B \in L$: $A \equiv B$.

An almost immediate corollary of this theorem given that $L \subseteq J$ and that J characterizes observational equivalence is that L also characterizes it.

Corollary 3.2. *If each R_e is image finite, then $L(p) = L(q)$ iff $p \, S \, q$.*

We could have gone further here and defined the set of modal conjunctive normal forms (or disjunctive normal forms) as a sublanguage of L and shown that every formula of J is semantically equivalent to a normal form formula. (The additional equivalences $\langle e \rangle(A \vee B) \equiv \langle e \rangle A \vee \langle e \rangle B$ and $[e](A \wedge B) \equiv [e]A \wedge [e]B$ are needed to show this.) However, the notion of modal normal form here, unlike the language L, is not very perspicuous.

4. A complete modal proof theory for a simple nondeterministic language

In this section we offer a complete (and sound) modal proof theory for a simple nondeterministic language of processes (a small subset of CCS [11, 13]). Let E, the set of atomic experiments, be a set of unary atomic action operators, NIL be a

nullary operator and + a nondeterministic dyadic operator. The set of processes NP is the least set such that

$$\mathrm{NIL} \in \mathrm{NP},$$

$$e.p \in \mathrm{NP} \quad \text{whenever } e \in E,$$

$$p + q \in \mathrm{NP} \quad \text{whenever } p, q \in \mathrm{NP}.$$

This little language is intended to capture when a process p can evolve to another process by performing an action in E: the idea is, for example, that $e.q$ becomes q by performing e. This performing of actions is defined using a labelled transition relation $\xrightarrow{e}$, $e \in E$, between processes which is the least relation such that:

$$e.q \xrightarrow{e} q,$$

$$p + q \xrightarrow{e} p' \quad \text{whenever } p \xrightarrow{e} p',$$

$$p + q \xrightarrow{e} q' \quad \text{whenever } q \xrightarrow{e} q'.$$

The notion of performing an action here is the correlate of being experimented on: p evolves to q by performing e ($p \xrightarrow{e} q$) in response to an e-experiment. If p cannot perform e then p cannot respond to an e-experiment: for instance, NIL, the process which cannot do anything, is unable to respond to any experiment in E. Consequently, the nondeterminism captured by + may be controlled by an observer (experimenter): for example, $e.p + b.q$ becomes p in response to an e-experiment, and q in response to a b-experiment. On the other hand, this is not true of $e.p + e.q$: an observer has no control as to whether p or q results in response to an e-experiment.

Two processes, p and $q \in \mathrm{NP}$, are said to be *equivalent*, written as $p \sim q$, just in case they are observationally equivalent. (Alternative equivalences based on experiments are proposed in [3, 5, 7, 9, 14].) The equivalence $\sim$ is a congruence: this result is proved in [8] where it is algebraically axiomatized. Clearly, for any e, $\xrightarrow{e}$ is image finite. Hence, by Corollary 3.2 we know that L characterizes $\sim$: two processes in NP are observationally distinguishable iff they are modally distinguishable. For instance, $p = e.(a.p' + b.q')$ satisfies $[e]\langle a\rangle\mathbf{true}$ whereas $q = e.a.p' + e.b.q'$ does not.

We now offer a sound and complete modal proof theory NL defined on L for the language of processes NP. We use a proof-theoretic relation $\vdash_{\mathrm{NL}}$ between processes and formulas: $p \vdash_{\mathrm{NL}} A$ means that $A \in L$ is provable of $p \in \mathrm{NP}$. The index NL is suppressed for the most part. Some of the following rules are analogous to Gentzen's introduction rules [4]: here, however, two very different kinds of objects are elements of either side of $\vdash$.

The system NL

true	$p \vdash \mathbf{true}$,
Nil	$\mathrm{NIL} \vdash [e]A$,
Act	$e.q \vdash [a]A$ whenever $e \neq a$,
$\vee I$	$\dfrac{p \vdash A}{p \vdash A \vee B} \quad \dfrac{p \vdash B}{p \vdash A \vee B}$,
$\wedge I$	$\dfrac{p \vdash A \quad p \vdash B}{p \vdash A \wedge B}$,
$\langle e \rangle I$	$\dfrac{p \vdash A}{e.p \vdash \langle e \rangle A}$,
$[e]I$	$\dfrac{p \vdash A}{e.p \vdash [e]A}$,
$\langle e \rangle + I$	$\dfrac{p \vdash \langle e \rangle A}{p+q \vdash \langle e \rangle A} \quad \dfrac{q \vdash \langle e \rangle A}{p+q \vdash \langle e \rangle A}$,
$[e] + I$	$\dfrac{p \vdash [e]A \quad q \vdash [e]A}{p+q \vdash [e]A}$.

The *rules* of NL are straightforward. $\mathrm{NIL} \vdash [e]A$ because NIL cannot perform any action in E. Similarly, $e.q \vdash [a]A$ whenever $a \neq e$ because $e.q$ can only perform e. This also accounts for the $\langle e \rangle I$ and $[e]I$ rules. The $\vee I$ and $\wedge I$ rules are as expected. The final two clauses of the labelled transition relation above account for the $\langle e \rangle + I$ rules: if p (or q) can evolve to a process which satisfies A by performing e then so can $p+q$. Similarly for $[e]+I$ which says that if every e-experiment on p and on q results in a process satisfying A then likewise for every e-experiment on $p+q$.

A *proof* in NL can be represented either as a finite tree, as in standard Gentzen proof systems, or as a finite sequence as in axiomatic proof systems. We give two example NL proofs:

$$a.p + b.q \vdash (\langle a \rangle \mathbf{true} \wedge \langle b \rangle \mathbf{true}) \wedge [c]\mathbf{false}$$

$$\langle a \rangle I \frac{p \vdash \mathbf{true}}{} \qquad \langle b \rangle I \frac{q \vdash \mathbf{true}}{}$$

$$\langle a \rangle + I \frac{a.p \vdash \langle a \rangle \mathbf{true}}{} \qquad \langle b \rangle + I \frac{b.q \vdash \langle b \rangle \mathbf{true}}{}$$

$$\wedge I\frac{a.p+b.q\vdash\langle a\rangle\mathbf{true}\quad a.p+b.q\vdash\langle b\rangle\mathbf{true}}{+I\dfrac{a.p\vdash[c]\mathbf{false}\quad b.q\vdash[c]\mathbf{false}}{a.p+b.q\vdash[c]\mathbf{false}}}[c]$$

$$\wedge I\frac{a.p+b.q\vdash\langle a\rangle\mathbf{true}\wedge\langle b\rangle\mathbf{true}\quad a.p+b.q\vdash[c]\mathbf{false}}{a.p+b.q\vdash(\langle a\rangle\mathbf{true}\wedge\langle b\rangle\mathbf{true})\wedge[c]\mathbf{false}}$$

$a.b\mathrm{NIL}+a.b.c.\mathrm{NIL}\vdash[a]\langle b\rangle[c]\mathbf{true}$

(i)	$\mathrm{NIL}\vdash[c]\mathbf{true}$	Nil *ax*,
(ii)	$b.\mathrm{NIL}\vdash\langle b\rangle[c]\mathbf{true}$	$\langle b\rangle I$,
(iii)	$a.b.\mathrm{NIL}\vdash[a]\langle b\rangle[c]\mathbf{true}$	$[a]I$,
(iv)	$\mathrm{NIL}\vdash\mathbf{true}$	true *ax*,
(v)	$c.\mathrm{NIL}\vdash[c]\mathbf{true}$	$[c]I$,
(vi)	$b.c.\mathrm{NIL}\vdash\langle b\rangle[c]\mathbf{true}$	$\langle b\rangle I$,
(vii)	$a.b.c.\mathrm{NIL}\vdash[a]\langle b\rangle[c]\mathbf{true}$	$[a]I$,
(viii)	$a.b.\mathrm{NIL}+a.b.c.\mathrm{NIL}\vdash[a]\langle b\rangle[c]\mathbf{true}$	$[a]+I$ on (iii) and (vii).

The system NL is both sound and complete. This is the content of the following theorem.

Theorem 4.1. $p\models_L A$ *iff* $p\vdash_{\mathrm{NL}}A$.

An obvious corollary is that $p\sim q$ iff $\mathrm{NL}(p)=\mathrm{NL}(q)$ where $\mathrm{NL}(p)=\{A:p\vdash_{\mathrm{NL}}A\}$. The system NL does not include a rule of the form

$$+I\qquad\frac{p\vdash A\quad q\vdash B}{p+q\vdash A*B}$$

where $*$ is a truth-functional connective on L. If the rule is to be sound then $*$ cannot be truth-functional unless $A*B\equiv\mathbf{true}$ for every A and B. This result, which is proved for the more general language J, is the content of the next theorem.

Theorem 4.2. *If* $*$ *is any truth-functional dyadic connective on* J *such that* $\exists C,D:C*D\not\equiv\mathbf{true}$, *then every rule of the form* $+I$ *above is unsound.*

Similarly, a more restricted rule of the form

$$+I\qquad\frac{p\vdash_J A\quad q\vdash_J A}{p+q\vdash_J A}$$

is also unsound: for example, let A be $\neg(\langle a\rangle\mathbf{true}\wedge\langle b\rangle\mathbf{true})$, then $a.\mathrm{NIL}\models_J A$ and $b.\mathrm{NIL}\models_J A$ but $a.\mathrm{NIL}+b.\mathrm{NIL}\not\models_J A$. Here, A is a formula whose content is negative in the sense that it says of a process what it cannot do. It is this which causes the problems. (It is unlikely that a language which does not express such negative information could characterize observational equivalence.) This discussion brings out that L is technically simpler for constructing a modal proof system for NP than the language J: there would be no simple $\neg I$ rule but instead introduction rules for $\neg A$ which depend upon the structure of A as evidenced by the NIL, Act, $\vee I$, $[e]I$ and $[e]+I$ rules of NL.

5. Complete modal proof theories for a nondeterministic language involving unobservable actions

In the case of the nondeterministic language of processes, NP, above every e-action is observable. This means that the evolving of one process to another is always observable. Suppose an unobservable action, a 'silent' action, τ is also included: Let $E' = E \cup \{\tau\}$. We call the resulting nondeterministic language NP'. There is no atomic experiment corresponding to τ. The notion of performing an observable action, an e-action, is redefined to take account of τ [8]. Let $p \stackrel{\tau^*}{\rightarrow} q$ stand for $\exists n \geq 0, p_1, \ldots, p_n$: $p \stackrel{\tau}{\rightarrow} p_1 \stackrel{\tau}{\rightarrow} \cdots \stackrel{\tau}{\rightarrow} p_n = q$; that is, $p \stackrel{\tau^*}{\rightarrow} q$ means that p can evolve to q by performing a finite number of unobservable actions. When $n = 0$ then $p = q$. Let $\stackrel{e}{\Rightarrow}, e \in E$, be a labelled transition relation on NP' defined as follows: $p \stackrel{e}{\Rightarrow} q$ iff $\exists p', p''$ s.t. $p \stackrel{\tau^*}{\rightarrow} p' \stackrel{e}{\rightarrow} p'' \stackrel{\tau^*}{\rightarrow} q$. The new relation $\stackrel{e}{\Rightarrow}, e \in E$, may absorb any finite sequence of unobservable actions before and after the action e. Note that $\stackrel{e}{\Rightarrow}$ is image finite in the case of NP'.

Although there is no experiment corresponding to τ its presence in a process can affect its observable behaviour. For example, consider $p = a.(b.\mathrm{NIL} + \tau.\mathrm{NIL})$ and $q = a.(b.\mathrm{NIL} + \mathrm{NIL})$ which is like p except for the absence of τ: p is observationally distinguishable from q because p can evolve to NIL whereas q can only evolve to $b.\mathrm{NIL} + \mathrm{NIL}$ in response to an a-experiment. Let $\approx$ be the observational equivalence on NP' defined using $\stackrel{e}{\Rightarrow}, e \in E$, to stand for R_e of the definition of Section 1 instead of $\stackrel{e}{\rightarrow}$ as in the case of NP. The equivalence $\approx$ is not a congruence: program contexts do not preserve it. For instance, $\tau.\mathrm{NIL} \approx \mathrm{NIL}$ but, as just noted, $a.(b.\mathrm{NIL} + \tau.\mathrm{NIL}) \not\approx a.(b.\mathrm{NIL} + \mathrm{NIL})$. Let $\approx_c$ be the largest congruence contained in $\approx$. This congruence is axiomatized algebraically in [8].

From the present discussion and Corollary 3:2 we know that the modal language L characterizes $\approx$. Below, we offer a sound and complete proof system UNL on L for NP'. We also show that $\approx_c$ can be characterized by an extension of L which we call L'. Furthermore, a sound and complete proof system UNL' on L' is given for NP'.

The NL rule $[e]I$ is not sound for NP': for example, $a.\mathrm{NIL} + \tau.\mathrm{NIL} \models_L \langle a \rangle \mathbf{true}$ but $e.(a.\mathrm{NIL} + \tau.\mathrm{NIL}) \not\models_L [e]\langle a \rangle \mathbf{true}$ because $e.(a.\mathrm{NIL} + \tau.\mathrm{NIL}) \stackrel{e}{\Rightarrow} \mathrm{NIL}$. However, it appears that a complete proof system for NP' on L should include a $[e]$ introduction rule to derive valid instances such as $e.(a.\mathrm{NIL} + \tau.a.\mathrm{NIL}) \models_L [e]\langle a \rangle \mathbf{true}$. In contrast, the NL $\langle e \rangle I$ rule is no longer sufficient: for example, $e.(a.\mathrm{NIL} + \tau.\mathrm{NIL}) \models_L \langle e \rangle [a] \mathbf{false}$ could not be derived using it because $a.\mathrm{NIL} + \tau.\mathrm{NIL} \not\models_L [a] \mathbf{false}$. This means that $\langle e \rangle$ and $[e]$ introduction rules have to be less 'local' than in NL. We achieve this by making use of contexts with the result that the proof systems below are less elegant than NL.

Let $\mathscr{C}$ be *the least set of contexts* on NP' such that

$\# \in \mathscr{C}$,

$\tau.\phi \in \mathscr{C}$ whenever $\phi \in \mathscr{C}$,

$p + \phi,\ \phi + p \in \mathscr{C}$ whenever $\phi \in \mathscr{C}$ and $p \in \mathrm{NP}'$.

When $\phi \in \mathscr{C}$, $\phi(p)$ stands for the process which is the result of replacing p for $\#$ in ϕ: for instance, if ϕ is $\tau.(p+\#)$, then $\phi(q)$ is $\tau.(p+q)$. The definition of $\mathscr{C}$ has the following consequence where τ^* stands for a finite—possibly empty—sequence of τ occurrences.

Lemma 5.1. $\tau^*.e.p \stackrel{e}{\Rightarrow} q$ *iff* $q = p$ *or* $\exists \phi \in \mathscr{C}$ s.t. $p = \phi(\tau.q)$.

Note the essential occurrence of τ in $\phi(\tau.q)$ in this lemma. An immediate corollary is the following.

Corollary 5.2

(i) $\tau^*.e.p \models_L \langle e\rangle A$ *iff* $p \models_L A$ *or* $\exists q, \phi$ s.t. $p = \phi(\tau.q)$ *and* $q \models_L A$

(ii) $\tau^*.e.p \models_L [e]A$ *iff* $p \models_L A$ *and* $\forall q, \phi$: *if* $p = \phi(\tau.q)$ *then* $q \models_L A$

The set $\{q: \exists \phi$ s.t. $p = \phi(\tau.q)\}$ is finite for each $p \in \mathrm{NP}'$. It is this corollary which justifies the $\langle e\rangle \phi I$ and $[e]\phi I$ rules below of UNL and the $[e]\phi I$ rule of UNL′.

The system UNL

The **true** $\bigvee I$, and $\bigwedge I$ rules of UNL below are as in NL. The Nil, Act, $\langle e\rangle I$, $\langle e\rangle + I$ and $[e] + I$ rules are similar to those of NL except that τ^* is prefaced to the process in the consequent or axiom. The significant differences to NL are the rules $\langle e\rangle \phi I$ and $[e]\phi I$ which we justified above. As before we suppress the index UNL.

true	$p \vdash$ **true**,
Nil	$\tau^*.\mathrm{NIL} \vdash [e]A$,
Act	$\tau^*.e.q \vdash [a]A$ when $e \neq a$,
$\bigvee I$	$\dfrac{p \vdash A}{p \vdash A \vee B} \quad \dfrac{p \vdash B}{p \vdash A \vee B}$,
$\bigwedge I$	$\dfrac{p \vdash A \quad p \vdash B}{p \vdash A \wedge B}$,
$\langle e\rangle I$	$\dfrac{p \vdash A}{\tau^*.e.p \vdash \langle e\rangle A}$,
$\langle e\rangle \phi I$	$\dfrac{p \vdash A}{\tau^*.e.\phi(\tau.p) \vdash \langle e\rangle A}$ for any context $\phi \in \mathscr{C}$,

$$[e]\phi I \quad \frac{p \vdash A, q_1 \vdash A, \ldots, q_n \vdash A}{\tau^*.e.p \vdash [e]A} \quad \text{where } \{q_1, \ldots, q_n\} = \{q : \exists \phi : p = \phi(\tau.q)\},$$

$$\langle e \rangle + I \quad \frac{p \vdash \langle e \rangle A}{\tau^*.(p+q) \vdash \langle e \rangle A} \quad \frac{q \vdash \langle e \rangle A}{\tau^*.(p+q) \vdash \langle e \rangle A},$$

$$[e] + I \quad \frac{p \vdash [e]A \quad q \vdash [e]A}{\tau^*.(p+q) \vdash [e]A}.$$

An example proof illustrating the use of the rules $\langle e \rangle \phi I$ and $[e]\phi I$ is as follows. Let

$$r = b.\mathrm{NIL} + \tau.\mathrm{NIL},$$

$$p = a.\mathrm{NIL},$$

$$q = \tau.p + \tau.(\tau.p + \tau.\tau.a.\tau.r).$$

The set $\{q : \exists \phi \text{ s.t. } q = \phi(\tau.q')\} = \{p, a.\tau.r, \tau.a.\tau.r, \tau.p + \tau.\tau.a.\tau.r\}$. We prove that $\tau.e.q \vdash [e]\langle a \rangle [b]\mathbf{false}$.

(i) $\mathrm{NIL} \vdash [b]\mathbf{false}$ Nil *ax*,

(ii) $\tau.p \vdash \langle a \rangle [b]\mathbf{false}$ $\langle a \rangle I$ on (i),

(iii) $q \vdash \langle a \rangle [b]\mathbf{false}$ $\langle a \rangle + I$ on (ii),

(iv) $p \vdash \langle a \rangle [b]\mathbf{false}$ $\langle a \rangle I$ on (i),

(v) $\tau.p + \tau.\tau.a.\tau.r \vdash \langle a \rangle [b]\mathbf{false}$ $\langle a \rangle + I$ on (ii),

(vi) $\tau.a.\tau.r \vdash \langle a \rangle [b]\mathbf{false}$ $\langle a \rangle \phi I$ on (i) (ϕ is $\tau.(b.\mathrm{NIL} + \not\#)$),

(vii) $a.\tau.r \vdash \langle a \rangle [b]\mathbf{false}$ $\langle a \rangle \phi I$ on (i) (ϕ is $\tau.(b.\mathrm{NIL} + \not\#)$),

(viii) $\tau.e.q \vdash [e]\langle a \rangle [b]\mathbf{false}$ $[e]\phi I$ on (iii), (iv), (v), (vi), and (vii).

The system UNL is both sound and complete.

Theorem 5.3. $\models_L A$ *iff* $p \vdash_{\mathrm{UNL}} A$.

The language L'

The equivalence $\approx_c$ was stipulated to be the largest congruence contained in $\approx$. (Recall that $\approx$ is not a congruence.) This congruence is more refined than $\approx$: for instance $\tau.\mathrm{NIL} \not\approx_c \mathrm{NIL}$. A first suggestion, then for a language which characterizes $\approx_c$ instead of $\approx$ is to extend L by the addition of a modal operator $\langle \tau \rangle$ corresponding to the unobservable action τ. This, in effect, allows us to talk directly about the silent moves of a process: $\tau.\mathrm{NIL} \models \langle \tau \rangle \mathbf{true}$ whereas $\mathrm{NIL} \not\models \langle \tau \rangle \mathbf{true}$. However, the resulting modal language would be too strong if iteration of such an operator is

allowed because $\tau.\tau.p \approx_c \tau.p$. Similarly, if it can appear within the scope of the other modal operators $[e]$ and $\langle e\rangle$ because, for instance, $e.\tau.p \approx_c e.p$. Consequently, we add to L a monadic modal operator O which cannot appear within the scope of any other modal operator including itself. Let L' be the *resulting modal language*. OOA, $\langle e\rangle(\langle b\rangle\mathbf{true} \wedge O\,\text{true})$ are not formulas of L' whereas $O\,\mathbf{true}$ and $\langle e\rangle\mathbf{true} \vee O\langle e\rangle\mathbf{true}$ are. The satisfaction relation for L' is as before except for the additional clause where $\stackrel{\tau}{\Rightarrow}$ is like $\stackrel{e}{\Rightarrow}$:

$$p \models_{L'} OA \text{ iff } \exists p' \text{ s.t. } p \stackrel{\tau}{\Rightarrow} p' \text{ and } p' \models_{L'} A.$$

The following theorem shows that L' characterizes $\approx_c$.

Theorem 5.4. *$p \approx_c q$ iff $L'(p) = L'(q)$.*

Note that this result unlike the characterization results of Sections 2 and 3 is process language dependent: it may fail to hold if additional operators are added to NP'. The process $\tau.p$ will, in general, be distinguishable from p by a formula of the form OA: for example, $\tau.\mathrm{NIL} \models_{L'} O\,\mathbf{true}$ whereas $\mathrm{NIL} \not\models_{L'} O\,\mathbf{true}$.

The system UNL'

We now offer a complete proof system UNL' on L'. The additional operator O allows us to dispense with the $\langle e\rangle\phi I$ rule (of UNL) but not with the $[e]\phi I$. (The latter rule is dispensable if the dual of O is also included as an operator.) We adopt the following convention: if $A \in L'$, then $A^* \in L$ is the formula which results from removing all occurrences of O in A.

true, Nil, Act, $\vee I$, $\wedge I$, $\langle e\rangle + I$, and $[e] + I$ rules as in UNL,

$\langle e\rangle I$ $\quad \dfrac{p \vdash A}{\tau^*.e.p \vdash \langle e\rangle A^*}$,

$[e]\phi I$ $\quad \dfrac{p \vdash A,\ q_1 \vdash A,\ \ldots,\ q_n \vdash A}{\tau^*.e.p \vdash [e]A^*}$ where $\{q_1, \ldots, q_n\} = \{q : \exists \phi.p = \phi(\tau.q)\}$,

OI $\quad \dfrac{p \vdash A}{\tau^*.\tau.p \vdash OA}$ where A does not contain O,

$O+I$ $\quad \dfrac{p \vdash OA}{\tau^*.(p+q) \vdash OA} \quad \dfrac{q \vdash OA}{\tau^*.(p+q) \vdash OA}$.

The following is a derived rule of UNL' for any $A \in L$ and $\phi \in \mathscr{C}$:

$$\phi I \frac{p \vdash A}{\phi(\tau.p) \vdash OA}.$$

It follows straightforwardly from OI and $O+I$. Thus, the $\langle e\rangle\phi I$ is derivable by application of $\langle e\rangle I$ to the consequent of ϕI. An example proof $e.(a.\mathrm{NIL}+\tau.b.\mathrm{NIL})\vdash\langle e\rangle[a]\mathbf{false}$ illustrates that $\langle e\rangle\phi I$ is derivable in UNL′:

(i) $b.\mathrm{NIL}\vdash[a]\mathbf{false}$ Act,

(ii) $\tau.b.\mathrm{NIL}\vdash O[a]\mathbf{false}$ OI,

(iii) $a.\mathrm{NIL}+\tau.b.\mathrm{NIL}\vdash O[a]\mathbf{false}$ $O+I$,

(iv) $e.(a.\mathrm{NIL}+\tau.b.\mathrm{NIL})\vdash\langle e\rangle[a]\mathbf{false}$ $\langle e\rangle I$.

The system UNL′ is both sound and complete.

Theorem 5.5. $p\models_{L'}A$ *iff* $p\vdash_{\mathrm{UNL}'}A$.

6. Concluding remarks: Proof systems for richer process languages

We have shown that complete proof systems can be given for non-deterministic languages of processes. Completeness is, here, relative to observational equivalence or congruence. However, the languages of processes involved are not very rich.

Following [8] a binary parallel operator | may be added to the nondetermistic process language. | is chosen to represent the combination of a pair of programs which may proceed concurrently and also communicate with each other. To express communication let $\bar{E}$ be a set of (observable) actions disjoint from E but in bijection with it. The bijection and its inverse are represented by $\bar{}$. Thus, $\bar{e}\in\bar{E}$ and $\bar{\bar{e}}=e$ whenever $e\in E$. Communication between two processes may occur when one admits an e-experiment and the other an $\bar{e}$-experiment: the resulting communication is represented by a τ-action. (Internal communication is therefore treated as unobservable.) The labelled transition relation for | is as follows where $e\in E\cup\bar{E}\cup\{\tau\}$:

$p|q\xrightarrow{e}p'|q$ whenever $p\xrightarrow{e}p'$,

$p|q\xrightarrow{e}p|q'$ whenever $e\xrightarrow{e}q'$,

$p|q\xrightarrow{\tau}p'|q'$ whenever $p\xrightarrow{e}p'$ and $q\xrightarrow{\bar{e}}q'$.

Defining the experiment relations $\{\overset{e}{\Rightarrow}: e\in E\cup\bar{E}\}$ as in the previous section gives rise to an observational equivalence $\approx'$ and a congruence $\approx'_c$. As before, these are different. Clearly, the modal language L characterizes $\approx'$. It appears that L' characterizes $\approx'_c$. Unfortunately, we have not discovered a natural proof system for this language of processes which has | introduction rules. However, Hennessy and Milner show that | can be eliminated from any process in this language using the following expansion theorem where $\sum_{1\leq i\leq n}e_i.p_i=e_1.p_1+\cdots+e_n.p_n$ if $n>0$ and NIL otherwise [8].

Theorem 6.1. *If* p *is* $\sum_i e_i.p_i$ *and* q *is* $\sum_j a_j.q_j$, *then*

$$p|q \approx'_c \sum_i e_i.(p_i|q) + \sum_j a_j.(p|q_j) + \sum_{e_i=\bar{a}_j} \tau.(p_i|q_j).$$

Thus, we may add to our proof system UNL′ the following rule:

$$|I.\frac{p \vdash A}{q|r \vdash A} \quad \text{whenever } p \approx'_c q|r \text{ by Theorem 6.1.}$$

This, of course, is not a very satisfactory proof rule. Firstly, it is very indirect. And secondly, the elimination of | goes through because each process represents a finite behaviour.

An alternative suggestion for dealing with | (loosely based on [1]) is to introduce a relativized satisfaction relation $\models_A$ where A is a formula. The intended meaning of $p \models_A B$ is:

$$\forall q: \quad q \models A \text{ implies } q|p \models B.$$

Thus, A in $\models_A$ represents, in some sense, an environment. Moreover, because | is associative $p \models_A B$ and $q \models_B C$ imply $p|q \models_A C$.

This means that we need to offer proof rules for a relativized proof-theoretic consequence relation $p \vdash_A B$. Of interest is that rules for | introduction will be analogous to Gentzen's cut rule [G]:

$$\frac{q \vdash A \quad p \vdash_A B}{q|p \vdash B},$$

$$\frac{q \vdash_A B \quad p \vdash_B C}{q|p \vdash_A C}.$$

A further extension to the language of processes is to add a facility for potentially infinite behaviours. In CCS this is achieved by the addition of process variables and a fix operator [11, 13]. Let X range over a set of process variables which are nullary operators. Also added is fix X, a monadic binding operator: in fix $X.p$ all free occurrences of X in p are bound by fix X. The labelled transition rule for fix is:

$$\text{fix } X.p \xrightarrow{e} p' \quad \text{whenever } p[\text{fix } X.p/X] \xrightarrow{e} p'$$

where $[\cdot/\cdot]$ denotes the usual notion of substitution. It is hoped that the addition of the following proof rule to the systems in the previous sections result in complete proof systems:

$$\text{fix } I \quad \frac{p[\text{fix } X.p/X] \vdash A}{\text{fix } X.p \vdash A}.$$

Acknowledgment

I would like to thank Stuart Anderson, Gerardo Costa, Rocco De Nicola, Matthew Hennessy and Robin Milner for helpful suggestions and interesting discussions, and Eleanor Kerse and Dorothy McKie for typing.

Appendix A

Proof of Theorem 4.1. We suppress the indices on the satisfaction and proof relation.

[$\Leftarrow$] *Soundness.* We show that the theorem holds for the axioms and is preserved by the rules.

true *ax* Clear because $p \models \mathbf{true}$.

Nil *ax* $\forall p, e: \langle \mathrm{NIL}, p\rangle \notin R_e$.

Hence $\mathrm{NIL} \models [e]A$.

Act *ax* If $p = e.q$, then $\forall r, a: a \neq e \Rightarrow \langle p, r\rangle \notin R_a$.

Hence $p \models [a]A$ when $a \neq e$.

$\vee I$ Suppose $p \models A$; then clearly $p \models A \vee B$.

Similarly, if $p \models B$, then $p \models A \vee B$.

$\wedge I$ Suppose $p \models A$ and $p \models B$; then $p \models A \wedge B$.

$\langle e\rangle I$ Suppose $p \models A$; then clearly $e.p \models \langle e\rangle A$.

$[e]I$ Suppose $p \models A$; then clearly $e.p \models [e]A$.

$\langle e\rangle + I$ Suppose $p \models \langle e\rangle A$; then $\exists p': \langle p, p'\rangle \in R_e$ and $p' \models A$.

By the + rule, if $p \xrightarrow{e} p'$, then $p + q \xrightarrow{e} p'$.

Hence, $p + q \models \langle e\rangle A$.

Similarly if $q \models \langle e\rangle A$.

$[e] + I$ Suppose $p \models [e]A$ and $q \models [e]A$.

Then $\forall p': \langle p, p'\rangle \in R_e \Rightarrow p' \models A$.

and $\forall q': \langle q, q'\rangle \in R_e \Rightarrow q' \models A$.

Hence $\forall r: \langle p + q, r\rangle \in R_e \Rightarrow r \models A$.

So $p + q \models [e]A$.

[$\Rightarrow$] *Completeness.* Induction on n (is the number of connectives in A).

Basis step ($n = 0$): $A = \mathbf{true}$. Clearly holds by the **true** axiom and because $p \not\models \mathbf{false}$.

Induction step $(n = k+1)$: *Case* $(A = B \vee C)$: $p \models A$ iff $p \models B$ or $p \models C$. Suppose $p \models B$ then by induction hypothesis $p \vdash B$. By $\bigvee I$, $p \vdash B \vee C$. Similarly if $p \models C$.

Case $(A = B \wedge C)$: $p \models A$ iff $p \models B$ and $p \models C$. By induction hypothesis $p \vdash B$ and $p \vdash C$. Hence $p \vdash B \wedge C$ by $\bigwedge I$.

Case $(A = \langle e \rangle B)$: Induction on structure of p.

Subcase $(p = \text{NIL})$: Impossible for $p \models A$.

Subcase $(p = b.q)$: Then $b = e$, otherwise $p \not\models A$. Hence, $q \models B$. By induction hypothesis, $q \vdash B$. By $\langle e \rangle I$ rule, $e.q \vdash \langle e \rangle B$.

Subcase $(p = q + r)$: Then $q \models \langle e \rangle B$ or $r \models \langle e \rangle B$. Suppose $q \models \langle e \rangle B$. By induction hypothesis $q \vdash \langle e \rangle B$. By rule $\langle e \rangle + I$, $p \vdash \langle e \rangle B$. Similarly if $r \models \langle e \rangle B$.

Case $(A = [e]B)$: Induction on structure of p.

Subcase $(p = \text{NIL})$: By Nil *ax.* $p \vdash [e]B$.

Subcase $(p = b.q)$: If $b \neq e$ then $p \vdash [e]B$ by Act *ax.* If $b = e$ then $q \models B$. By induction hypothesis $q \vdash B$. Hence by $[e]I$, $p \vdash [e]B$.

Subcase $(p = q + r)$: Then $q \models [e]B$ and $r \models [e]B$. By induction hypothesis $q \vdash [e]B$ and $r \vdash [e]B$. By $[e] + I$, $p \vdash [e]B$. □

Theorem 4.2. *If* $A, B \in J$, $p, q \in \text{NP}$ *and* $*$ *is a truth-functional dyadic operation such that* $C * D \not\equiv$ **true** *for some* $C, D \in J$, *then no rule of the form*

$$+I \qquad \frac{p \vdash A \quad q \vdash B}{p + q \vdash A * B}$$

is sound.

Proof. There are sixteen possible truth-functional dyadic operators. $A * B \equiv$ **true** is excluded and $A * B \equiv \neg$**true** is clearly unsound. We show that the other fourteen cases are excluded by the following examples

(1) $a.\text{NIL} \models [b]$**false**. $b.\text{NIL} \models [a]$**false**.

(2) $a.\text{NIL} \models \langle a \rangle$**true**. $b.\text{NIL} \models \langle b \rangle$**true**.

(3) $a.\text{NIL} \models \langle a \rangle$**true**. $b.\text{NIL} \models [a]$**false**.

Cases $A * B \equiv A \wedge B$, $A \wedge \neg B$, $\neg A \wedge B$, A, B, $A \vee B$ are excluded by (1).

Cases $A * B \equiv \neg A \wedge \neg B$, $\neg A$, $\neg B$, $(A \vee B) \wedge \neg(A \wedge B)$, $\neg A \vee \neg B$ are excluded by (2).

Cases $A * B \equiv \neg A \vee B$, $A \vee \neg B$, $(A \wedge B) \vee (\neg A \wedge \neg B)$ are excluded by (3). □

Lemma 5.1. $\tau^*.e.p \stackrel{e}{\Rightarrow} q$ *iff* $q = p$ or $\exists \phi \in \mathscr{C}$: $p = \phi(\tau.q)$.

Proof. [$\Rightarrow$] Consider the possible structure of q by induction on p.

(i) $p = \tau^*.\text{NIL}$: $q = p$ or $p = \tau.\tau^*.q$. The result follows by definition of the set $\mathscr{C}$.

(ii) $p = \tau^*.a.r$: $q = p$ or $p = \tau.\tau^*.q$. As before, the result follows.

(iii) $p = \tau^*.(s + r)$: $q = p$ or $p = \tau.\tau^*.q$ as before.

Otherwise, q is s.t. $s \xrightarrow{\tau.\tau^*} q$ or $r \xrightarrow{\tau.\tau^*} q$. Again the result follows by definition of $\mathscr{C}$.

[$\Leftarrow$] Suppose $q = p$; then clearly $\tau^*.e.p \stackrel{e}{\Rightarrow} p$. Otherwise induction on ϕ.

(i) $\phi = \#$. Then $p = \tau.q$. Clearly, $\tau^*.e.\tau.q \stackrel{e}{\Rightarrow} q$.

(ii) $\phi = \tau.\psi$. Then $p = \tau.\psi(\tau.q)$. Again $\tau^*.e.\tau.\psi(\tau.q) \stackrel{e}{\Rightarrow} q$ by induction hypothesis.

(iii) $\phi = r + \psi$ or $\psi + r$. Suppose $\phi = r + \psi$ then $p = r + \psi(\tau.q)$. By induction hypothesis $\tau^*.e.\psi(\tau.q) \stackrel{e}{\Rightarrow} q$. By rules for + and $\Rightarrow \tau^*.e.(r + \psi(\tau.q)) \stackrel{e}{\Rightarrow} q$. Similarly for $\phi = \psi + r$. □

Theorem 5.3. *$p \models_L A$ iff $p \vdash_{\mathrm{UNL}} A$.*

Proof. We suppress indices on both the proof relation and satisfaction relation. The proof is similar to that of Theorem 4.1 which we appeal to.

[$\Leftarrow$] *Soundness.* Clearly, the axioms are sound. The $\bigvee I$ and $\bigwedge I$ rules are sound by Theorem 4.1.

Case ($\langle e\rangle I$): Suppose $p \models A$; then clearly $\tau^*.e.p \models \langle e\rangle A$.

Case ($\langle e\rangle \phi I$): Suppose $p \models A$. Then $\tau^*.e.q \stackrel{e}{\Rightarrow} p$ iff $q = p$ or $\exists \phi.q = \phi(\tau.p)$ by Lemma 5.1. Therefore, $\tau^*.e.q \models \langle e\rangle A$.

Case ($[e]\phi I$): Suppose $p \models A$ and $q_1 \models A, \ldots, q_n \models A$. Consider the set $\{q' : \tau^*.e.q \stackrel{e}{\Rightarrow} q'\}$. If this set $= \{p, q_1, \ldots, q_n\}$, then $\tau^*.e.q \models [e]A$.

Case ($\langle e\rangle + I$): Suppose $p \models \langle e\rangle A$; then $\exists p' : p \stackrel{e}{\Rightarrow} p'$ and $p' \models A$. By rules for +, if $p \stackrel{e}{\Rightarrow} p'$, then $\tau^*.(p + q) \stackrel{e}{\Rightarrow} p'$, and the result follows. Similarly for $q \models \langle e\rangle A$.

Case ($[e] + I$): Suppose $p \models [e]A$ and $q \models [e]A$. Then $\forall p' : p \stackrel{e}{\Rightarrow} p'$ implies $p' \models A$ and $\forall q' : q \stackrel{e}{\Rightarrow} q'$ implies $q' \models A$. Thus $\forall r : \tau^*.(p + q) \stackrel{e}{\Rightarrow} r$ implies $r \models A$. The result follows.

[$\Rightarrow$] *Completeness.* Induction on n (is the number of connectives in A). The case $n = 0$ is as in Theorem 4.1. The subcases $A = B \vee C$, $A = B \wedge C$ of $n = k + 1$ are also as in Theorem 4.1.

Case ($A = \langle e\rangle B$): Induction on p.

Subcase ($p = \tau^*.\mathrm{NIL}$): Impossible.

Subcase ($p = \tau^*.b.q$): Then $b = e$. By Corollary 5.2, $q \models B$ or $\exists r, \phi$: $q = \phi(\tau.r) \wedge r \models B$. These cases are covered by the rules $\langle e\rangle I$ and $\langle e\rangle \phi I$.

Subcase ($p = \tau^*.(q + r)$): Then $q \models \langle e\rangle B$ or $r \models \langle e\rangle B$. The result follows by $\langle e\rangle + I$ rules.

Case ($A = [e]B$): Induction on p.

Subcase ($p = \tau^*.\mathrm{NIL}$): By NIL *ax*, $p \vdash [e]B$.

Subcase ($p = \tau^*.b.q$): If $b \neq e$, then $p \vdash [e]B$ by Act *ax*. If $b = e$, then $p \models [e]B$ iff $q \models B$ and $\forall r, \phi$ *s.t.* $q = \phi(\tau.r)$ implies $r \models B$.

The result follows by $[e]\phi I$ rule.

Subcase ($p = \tau^*.(q + r)$): Then $q \models [e]B$ and $r \models [e]B$. The result follows by the $[e] + I$ rule. □

For the proof of Theorem 5.4 we inductively define the *modal degree* of a formula $A \in L'$, written as $m(a)$:

$$m(\mathbf{true}) = m(\mathbf{false}) = 0,$$

$$m(A \wedge B) = m(A \vee B) = \max\{m(A), m(B)\},$$

$$m(\langle e\rangle A) = m([e]A) = m(OA) = m(A) + 1.$$

We let L'_n stand for the language $\{A : A \in L' \wedge m(A) \leq n\}$.

We also define the *set of general contexts* on NP′ as the least set D such that

(i) $\# \in D$.

(ii) $\tau.\psi,\ e.\psi \in D$ whenever $\psi \in D$.

(iii) $p + \psi,\ \psi + p \in D$ whenever $\psi \in D$ and $p \in \mathrm{NP}'$.

Hence we define $p \approx_c q$ as $\forall \psi \in D\colon \psi(p) \approx \psi(q)$.

In the following theorem we make use of induction of the number of 'constructs' in ψ which is defined inductively:

(i) $c(\#) = 0$.

(ii) $c(\tau.\psi) = c.(e.\psi) = 1 + c(\psi)$.

(iii) $c(p + \psi) = c(\psi + p) = 2 + c(\psi)$.

Theorem 5.4. *$p \approx_c q$ iff $L'(p) = L'(q)$.*

Proof. It suffices to show that $(\forall \psi \in D\colon \psi(p) \approx_n \psi(q))$ iff $L'_n(p) = L'_n(q)$.

(1) $n = 0$:

$$\psi(p) \approx_0 \psi(q)\ (\forall p, q, \psi) \text{ and } p \models \mathbf{true}\ (\forall p) \text{ and } p \not\models \mathbf{false}\ (\forall p).$$

(2) $(\forall \psi \in D\colon \psi(p) \approx_{n+1} \psi(q)) \Rightarrow L'_{n+1}(p) = L'_{n+1}(q)$: We know that $p \approx_{n+1} q \Rightarrow L_{n+1}(p) = L_{n+1}(q)$ via Theorem 2.1 and Corollary 3.2. Hence, $\forall \psi \in D$: $\psi(p) \approx_{n+1} \psi(q) \Rightarrow L_{n+1}(p) = L_{n+1}(q)$.

The only difference between L and L' is the operator O. We have to show that if $\forall \psi \in D\colon \psi(p) \approx_{n+1} \psi(q)$ then $p \models OA$ iff $q \models OA$.

Suppose $p \models OA$ and $q \not\models OA$. Then $\exists p'\colon p \stackrel{\tau}{\Rightarrow} p'$ and $p' \models A$ where $A \in L_n$. And $\forall q'\colon q \stackrel{\tau}{\Rightarrow} q'$ implies $q' \not\models A$. Let ψ be $e.(b.\mathrm{NIL} + \#)$ where b does not occur in p or in q. Then $\psi(p) \stackrel{e}{\Rightarrow} p'$ and $p' \models A$ and $\neg\exists s\colon p' \stackrel{b}{\Rightarrow} s$. But $\forall q'\colon \psi(q) \stackrel{e}{\Rightarrow} q'$ either $q' \not\models A$ or $\exists r\colon q' \stackrel{b}{\Rightarrow} r$. But $A \in L_n$. Hence $\psi(p) \not\approx_{n+1} \psi(q)$ which contradicts hypothesis.

(3) $\exists \psi\colon \psi(p) \not\approx_{n+1} \psi(q) \Rightarrow L'_{n+1}(p) \neq L'_{n+1}(q)$: Induction on ψ.

(i) $\psi = \#$: The result follows via Theorem 2.1 and Corollary 3.2 because $L \subseteq L'$.

(ii) $\psi = \tau.T$: $\tau.T(p) \not\approx_{n+1} \tau.T(q)$. But this holds iff $T(p) \not\approx_{n+1} T(q)$.

(iii) $\psi = e.T$: $e.T(p) \not\approx_{n+1} e.T(q)$. $\exists p' : T(p) \stackrel{\tau}{\Rightarrow} p'$ and $\forall q' : T(q) \stackrel{\tau}{\Rightarrow} q'$ implies $p' \not\approx_n q'$. Prove it by induction on $c(T)$.

Case $(c = 0)$: $p \stackrel{\tau}{\Rightarrow} p'$ and $\forall q' : q \stackrel{\tau}{\Rightarrow} q' p' \not\approx_n q'$. Thus $\exists A \in L : p \models OA$ and $q \not\models OA$.

Case $(c = k+1)$: *Subcase* $(T = \tau.\Omega)$: Follows by induction hypothesis.

Subcase $(T = b.\Omega)$: Clearly $e.T(p) \not\approx_{n+1} e.T(q)$ just in case $T(p) \not\approx_n T(q)$. By main induction hypothesis $L'_n(p) \neq L'_n(q)$, hence $L'_{n+1}(p) \neq L'_{n+1}(q)$.

Subcase $(T = r + \Omega, \Omega + r)$: Suppose $T = r + \Omega$. Then $e.\Omega(p) \not\approx_{n+1} e.\Omega(q)$ (because r won't make a difference) and $c(e.\Omega) < C(r + \Omega)$. Therefore, the result follows. Similarly for $T = \Omega + r$.

(4) $\psi = r + T, T + r$: Clearly, $r + T(p) \not\approx_{n+1} r + T(q)$ iff $T(p) \not\approx_{n+1} T(q)$. The result follows by induction hypothesis. Similarly for $T(p) + r$. □

Theorem 5.5. *$p \models_{L'} A$ iff $p \vdash_{\mathrm{UNL}'} A$.*

Proof. The proof is very close to that of Theorem 5.3 above.

[⇐] *Soundness.* Soundness of rules except for OI and $O + I$ is almost as in Theorem 5.3.

Case (OI): Suppose $p \models A$ and A does not contain O; then clearly $\tau^*.\tau.p \models OA$.

Case $(O + I)$: Suppose $p \models OA$; then $\exists p' : p \stackrel{\tau}{\Rightarrow} p' \wedge p' \models A$. Clearly $\tau^*(p + q) \stackrel{\tau}{\Rightarrow} p'$, thus $\tau^*(p + q) \models OA$. Similarly, if $q \models OA$.

[⇒] *Completeness.* Proof as in Theorem 5.3 except for a new case:

Case $(A = OB)$: $p \models A$ iff $\exists p' : p \stackrel{\tau}{\Rightarrow} p'$ and $p' \models B$, thus, by induction hypothesis, $p' \vdash B$, and by OI and $O + I$ rules the result follows. □

References

[1] H. Barringer and R. Kuiper, Towards the hierarchical, temporal logic, specification of concurrent systems, Res. Rept., Dept. of Computer Science, University of Manchester, 1983.

[2] S. Brookes and W. Rounds, Behaviour equivalence relations induced by programming logics, *Lecture Notes in Computer Science* **154** (Springer, Berlin, 1983) 97–108.

[3] R. de Nicola and M. Hennessy, Testing equivalences for processes, Res. Rept., Dept. of Computer Science, Edinburgh Univ., CSR-123-82. (Shortened version in *Lecture Notes in Computer Science* **154** (Springer, Berlin, 1982) 548–560.)

[4] G. Gentzen, Investigations into logical deduction, in: M. E. Szabo, ed., *The Collected Works of Gerhard Gentzen* (North-Holland, Amsterdam, 1969).

[5] M. Hennessy, Synchronous and asynchronous experiments on processes, Res. Rept., Dept. of Computer Science, Edinburgh Univ., CSR-125-82, 1982.

[6] M. Hennessy, Axiomatizing finite delay operators, Res. Rept., Dept. of Computer Science, Edinburgh Univ., CSR-124-82, 1982.

[7] M. Hennessy, A model for nondeterministic machines, Res. Rept., Dept. of Computer Science, Edinburgh Univ., 1983.

[8] M. Hennessy and R. Milner, On observing nondeterminism and concurrency, *Lecture Notes in Computer Science* **85** (Springer, Berlin, 1980) 255–309 (extended version to appear in *JACM*).

[9] C. Hoare, S. Brookes and A. Roscoe, A theory of communicating sequential processes, Tech. Monograph Prg-16, Computing Laboratory, Univ. of Oxford, 1981.

[10] M. Hennessy and C. Stirling, The power of the future perfect in program logics, Res. Rept., Dept. of Computer Science, Edinburgh Univ., CSR-156-83, 1983.

[11] R. Milner, *A Calculus of Communicating Systems*, Lecture Notes in Computer Science **92** (Springer, Berlin, 1980).

[12] R. Milner, A modal characterization of observable machine-behaviour, *Lecture Notes in Computer Science* **112** (Springer, Berlin, 1981).

[13] R. Milner, Calculi for synchrony and asynchrony, *Theoret. Comput. Sci.* **25**(3) (1983) 267–310.

[14] W. Rounds and S. Brookes, Possible futures, acceptances, refusals, and communicating processes, in: *Proc. 22nd Annual Symp. on Foundations of Computer Science*, Nashville, TN (1981) 140–149.